

# Selected Data Manipulation Techniques in R

Using R Studio



# Imported Your Data Already?

- If you already have your data in RStudio, you can skip the slides providing an overview of the import process.



# Dataset

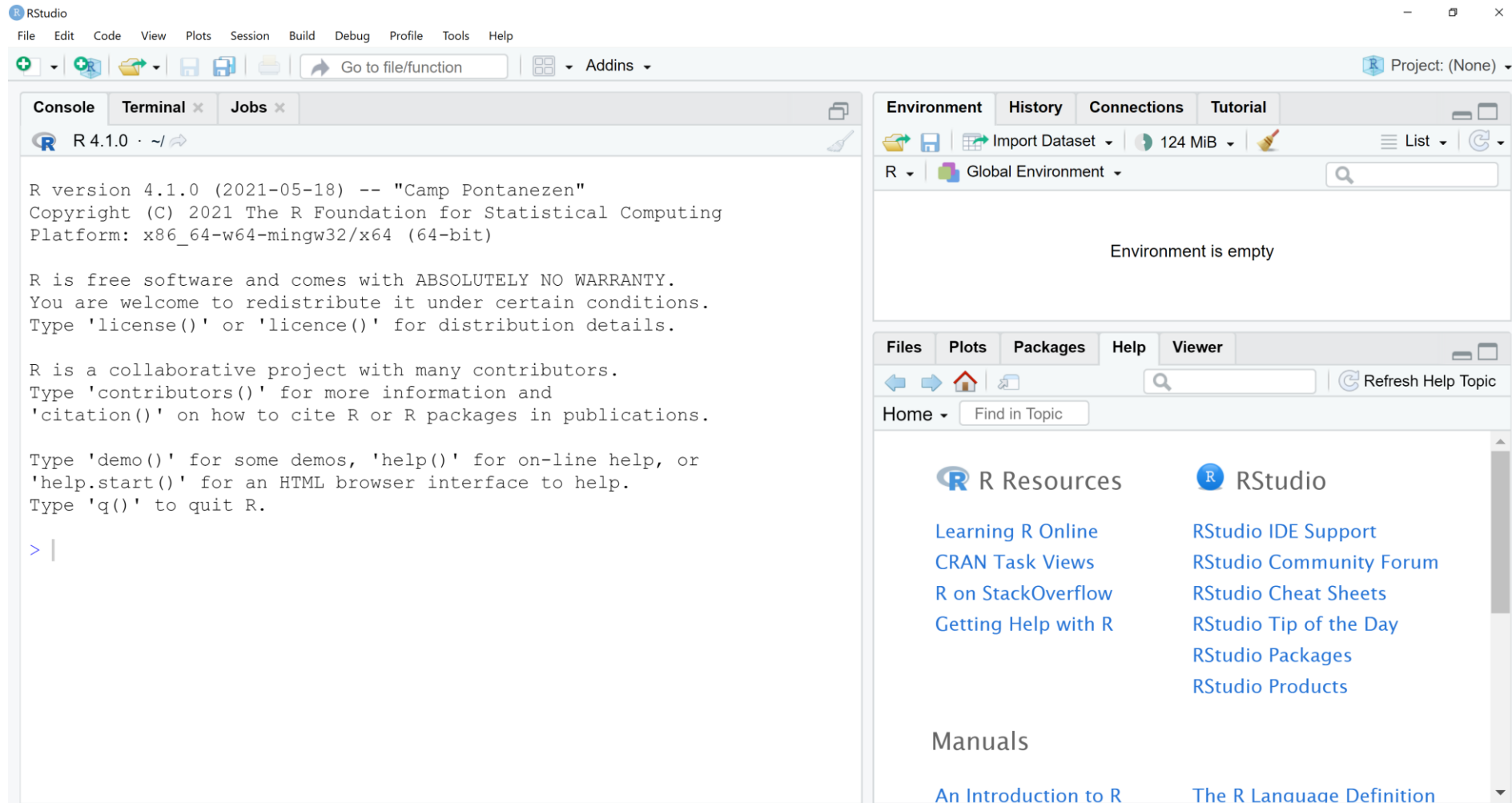
- This tutorial is a walkthrough with a sample set of data. You may use this to walk through the tutorial, if you wish, but for your assignments, you will be asked to use your own dataset (as specified within the course).

## Dataset reference:

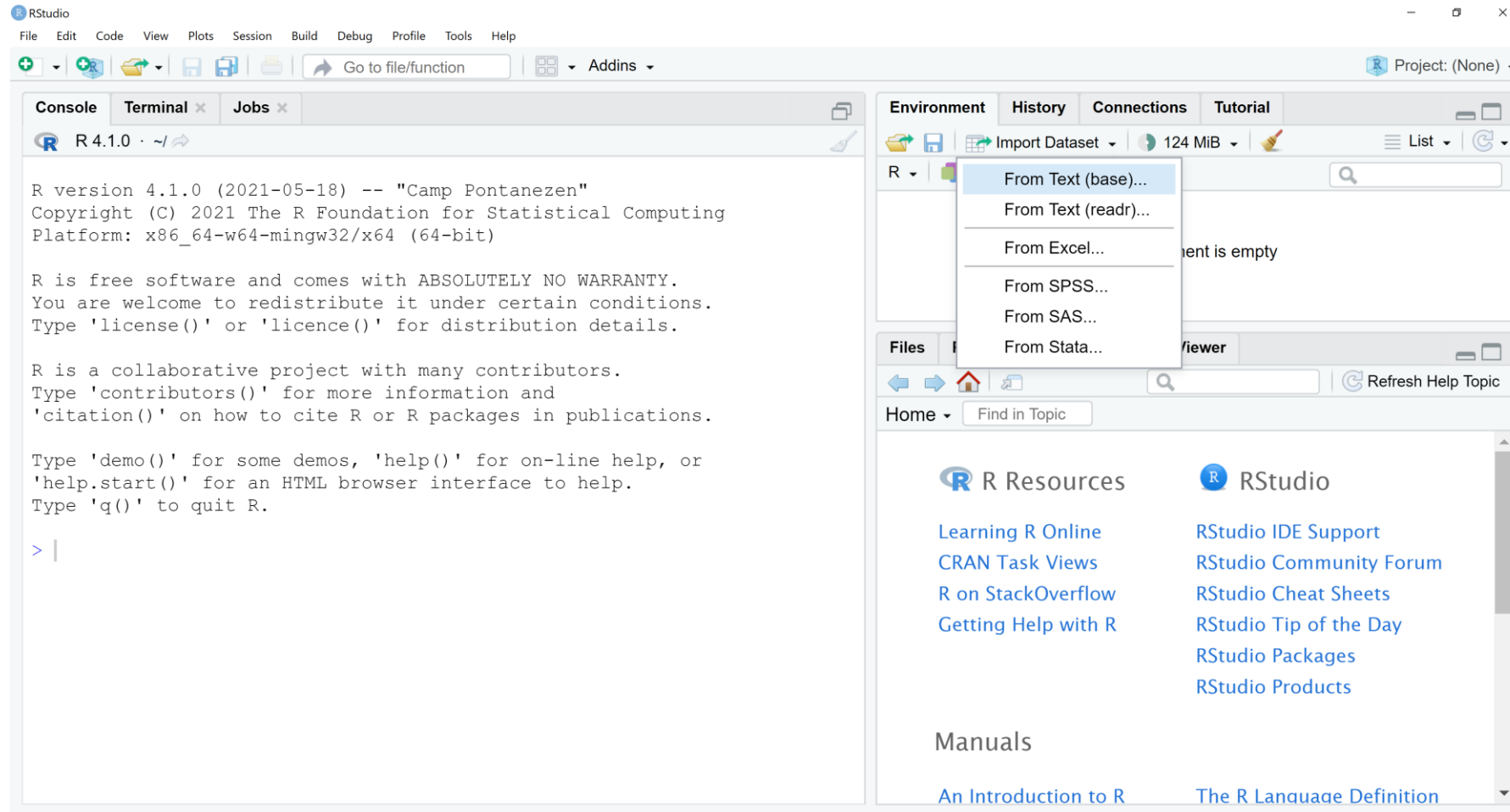
Skoryk, M. (2021). Sepsis Prediction from Clinical Data. Version 1.  
Retrieved from <https://www.kaggle.com/maxskoryk/datasepsis>



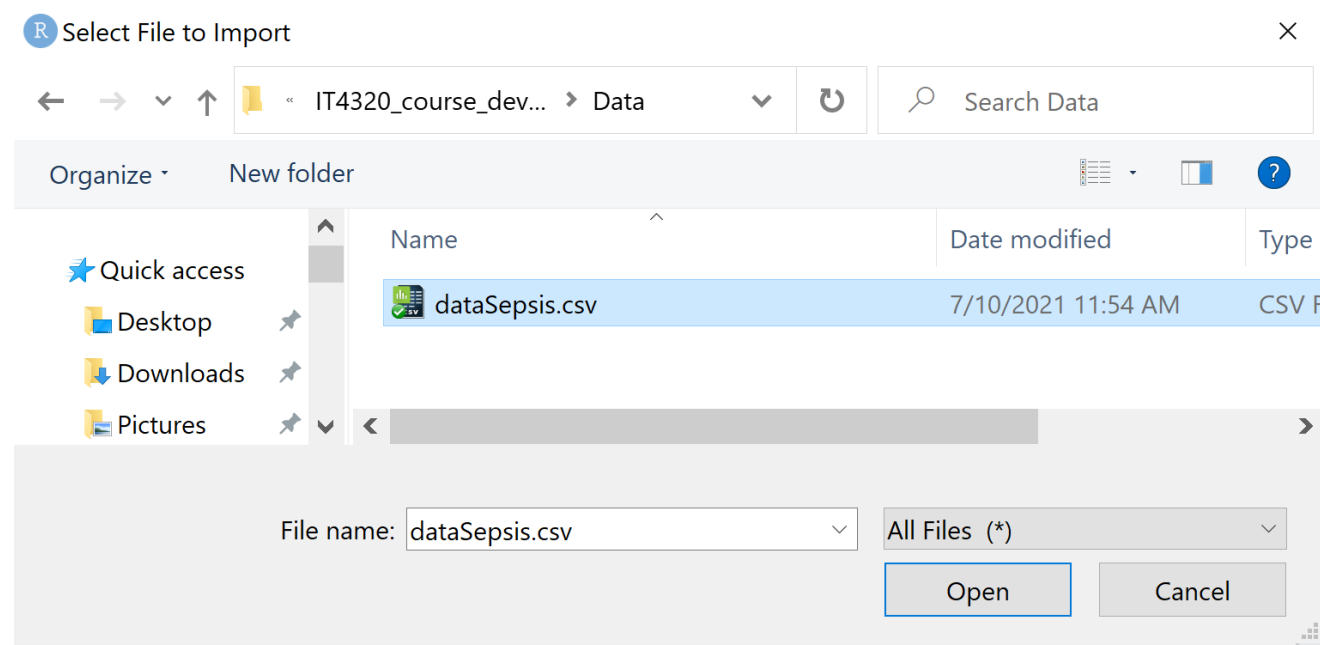
# Open R Studio



# Click on “Import Data” and Choose “From Text (base)”



# Navigate to Your Dataset, Then Click “Open”



# Select Options to Import Your Data Based on the Format of Your Text File

The screenshot shows the RStudio interface with the 'Import Dataset' dialog box open. The dialog has several sections: 'Name' (dataSepsis), 'Input File' (a text file path), 'Encoding' (Automatic), 'Heading' (No), 'Row names' (Automatic), 'Separator' (Semicolon), 'Decimal' (Period), 'Quote' (Double quote), 'Comment' (None), 'na.strings' (NaN), and 'Strings as factors' (unchecked). A 'Data Frame' preview is shown at the bottom. Annotations with blue brackets point to different parts of the dialog:

- Import Options:** Points to the 'Encoding', 'Heading', 'Row names', 'Separator', 'Decimal', 'Quote', 'Comment', 'na.strings', and 'Strings as factors' settings.
- Raw Data File Preview:** Points to the 'Input File' text area.
- "To Be Imported" Data File Preview:** Points to the 'Data Frame' table.

V1	V2	V3	V4	V5	V6	V7	V8	V9
HR	O2Sat	Temp	SBP	MAP	DBP	Resp	EtCO2	BaseExcess
103	90	NaN	NaN	NaN	NaN	30	NaN	2.3
58	95	36.11	143	77	47	11	NaN	Na
91	94	38.5	133	74	48	34	NaN	Na
92	100	NaN	NaN	NaN	NaN	NaN	NaN	Na
155.5	94.5	NaN	147.5	102	NaN	33	NaN	-2
73	99	36.06	100	67	49.5	16.5	NaN	-6
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0
82	100	35.5	112	79.5	63	14	NaN	0
89	100	NaN	141	85	57	17	NaN	1
100	95	37.28	121	20	NaN	NaN	NaN	Na
95	100	NaN	89	62.33	NaN	18	NaN	Na
86	96	38	111	66	49	17	NaN	1
88	100	36.3	99	66	52	16	NaN	-1



# The Options Displayed are Those Required to Successfully Import the sepsis dataset.

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Jobs

R 4.1.0 · ~/\

R version 4.1.0 (2021-05-25)  
Copyright (C) 2021 The R Foundation for Statistical Computing  
Platform: x86\_64-w64-mingw32

R is free software and you are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project. You can join the project by visiting the R project website.  
Type 'contributors()' for more information and 'citation()' on how to cite R in publications.

Type 'demo()' for some demos, 'help.start()' for an HTML browser interface to help, and 'help()' for on-line help with your current session.

Type 'q()' to quit R.

```
> dataSepsis <- read.csv("C:/Users/IT4320/Desktop/sepsis.csv", na.strings="NaN", as.is=TRUE)  
> View(dataSepsis)
```

Import Dataset

Name: dataSepsis

Input File: C:/Users/IT4320/Desktop/sepsis.csv

Encoding: Automatic

Heading: ☒ Yes ☐ No

Row names: Automatic

Separator: Semicolon

Decimal: Period

Quote: Double quote (")

Comment: None

na.strings: NaN

☒ Strings as factors

Data Frame

HR	O2Sat	Temp	SBP	MAP	DBP	Resp	EtCO2
103.0	90.0	NaN	NaN	NaN	NaN	30.0	NaN
58.0	95.0	36.11	143.0	77.00	47.0	11.0	NaN
91.0	94.0	38.50	133.0	74.00	48.0	34.0	NaN
92.0	100.0	NaN	NaN	NaN	NaN	NaN	NaN
155.5	94.5	NaN	147.5	102.00	NaN	33.0	NaN
73.0	99.0	36.06	100.0	67.00	49.5	16.5	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
82.0	100.0	35.50	112.0	79.50	63.0	14.0	NaN
89.0	100.0	NaN	141.0	85.00	57.0	17.0	NaN
100.0	95.0	37.28	121.0	20.00	NaN	NaN	NaN
95.0	100.0	NaN	89.0	62.33	NaN	18.0	NaN
86.0	96.0	38.00	111.0	66.00	49.0	17.0	NaN
88.0	100.0	36.30	99.0	66.00	52.0	16.0	NaN
116.0	97.0	38.28	200.0	108.00	90.0	24.0	NaN

Project: (None)

Refresh Help Topic

Studio

IDE Support

Community Forum

Cheat Sheets

Tip of the Day

Packages

Products

An Introduction to R

The R Language Definition





# Scroll Down and Click “Import” to Complete Import Process

The screenshot shows the RStudio interface with the 'Import File' dialog box open. The file name is 'dataSepsis'. The dialog box has the following settings:

- Encoding: Automatic
- Heading: ☒ Yes ☐ No
- Row names: Automatic
- Separator: Semicolon
- Decimal: Period
- Quote: Double quote (")
- Comment: None
- na.strings: NaN
- ☒ Strings as factors

The 'Data Frame' preview shows the following columns: HR, O2Sat, Temp, SBP, MAP, DBP, Resp, EtCO2. The data is as follows:

HR	O2Sat	Temp	SBP	MAP	DBP	Resp	EtCO2
103.0	90.0	NaN	NaN	NaN	NaN	30.0	NaN
58.0	95.0	36.11	143.0	77.00	47.0	11.0	NaN
91.0	94.0	38.50	133.0	74.00	48.0	34.0	NaN
92.0	100.0	NaN	NaN	NaN	NaN	NaN	NaN
155.5	94.5	NaN	147.5	102.00	NaN	33.0	NaN
73.0	99.0	36.06	100.0	67.00	49.5	16.5	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
82.0	100.0	35.50	112.0	79.50	63.0	14.0	NaN
89.0	100.0	NaN	141.0	85.00	57.0	17.0	NaN
100.0	95.0	37.28	121.0	20.00	NaN	NaN	NaN
95.0	100.0	NaN	89.0	62.33	NaN	18.0	NaN
86.0	96.0	38.00	111.0	66.00	49.0	17.0	NaN
88.0	100.0	36.30	99.0	66.00	52.0	16.0	NaN
116.0	97.0	38.28	200.0	108.00	90.0	24.0	NaN

The 'Import' button is highlighted with a blue arrow.



# You May Verify Successful Upload On the Following Screen

The screenshot displays the RStudio IDE with the 'dataSepsis' dataset loaded. The 'Environment' pane on the right shows the dataset with 36,302 observations and 41 variables. The 'Data' pane on the right provides a summary of the dataset. The 'Console' pane at the bottom shows the R script used to import the data. The 'Viewer' pane on the right displays R resources and manuals.

**Imported Data**

	HR	O2Sat	Temp	SBP	MAP	DBP	Resp	EtCO2	BaseExcess	HCC
1	103.0	90.0	NA	NA	NA	NA	30.0	NA	21.0	
2	58.0	95.0	36.11	143.00	77.00	47.0	11.0	NA	NA	
3	91.0	94.0	38.50	133.00	74.00	48.0	34.0	NA	NA	
4	92.0	100.0	NA	NA	NA	NA	NA	NA	NA	
5	155.5	94.5	NA	147.50	102.00	NA	33.0	NA	-12.0	
6	73.0	99.0	36.06	100.00	67.00	49.5	16.5	NA	-8.0	
7	NA	NA	NA	NA	NA	NA	NA	NA	0.0	

Showing 1 to 7 of 36,302 entries, 41 total columns

**Dataset Summary**

dataSepsis 36302 obs. of 41 variables

**Logfile Indicating Options in the Import Process**

```
R 4.1.0 ~/  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> dataSepsis <- read.csv("C:/Users/stefa/Dropbox/teaching/Teaching/Previous Courses/IT4320_course_development/dataSepsis.csv", header=FALSE, sep=";", na.strings="NaN", stringsAsFactors=TRUE)  
> View(dataSepsis)  
> dataSepsis <- read.csv("C:/Users/stefa/Dropbox/teaching/Teaching/Previous Courses/IT4320_course_development/dataSepsis.csv", sep=";", na.strings="NaN")
```



# More Options for Importing Data Into R Studio

<https://support.rstudio.com/hc/en-us/articles/218611977-Importing-Data-with-the-RStudio-IDE>



# Selected Data Manipulation Examples

- Checking the format of a variable
- Converting a numeric variable to a character variable
- Converting a character variable to a numeric variable
- Creating a calculated variable
- Populating values of a variable using an if-then statement
- Combining multiple character variables
- Taking only a portion of a character variable using a substring



# Checking the format of a variable

```
> is.numeric(dataSepsis$HR)
[1] TRUE
```

```
> is.character(dataSepsis$O2Sat)
[1] FALSE
```

format check function:  
Asks R if a variable is a specific  
format (numeric or character)

Variable to  
be checked

Result:

TRUE= the variable is the variable type  
specified in the format check function.  
FALSE= the variable is not the variable type  
specified in the format check function.



# Converting a numeric variable to a character variable

```
> dataSepsis$HR_char <- as.character(dataSepsis$HR)
```

New variable  
name

Export/Assign function:  
Tells R that it will be exporting the  
values from one variable to another  
variable

Function to  
convert to  
character

Old variable  
name



# Converting a character variable to a numeric variable

```
> dataSepsis$HR_num <- as.numeric(dataSepsis$HR_char)
```

New variable  
name

Export/Assign function:  
Tells R that it will be exporting the  
values from one variable to another  
variable

Function to  
convert to  
numeric

Old variable  
name



# Creating a calculated variable

```
> dataSepsis$Unitsum <- dataSepsis$Unit1 + dataSepsis$Unit2
```

New  
variable  
name

Export/Assign function:  
Tells R that it will be exporting  
the values from one variable  
to another variable

Variable 1  
to be  
added

Mathematical operator  
(This can be any usual  
mathematical operator (+ - /  
\* ^ for a few examples)

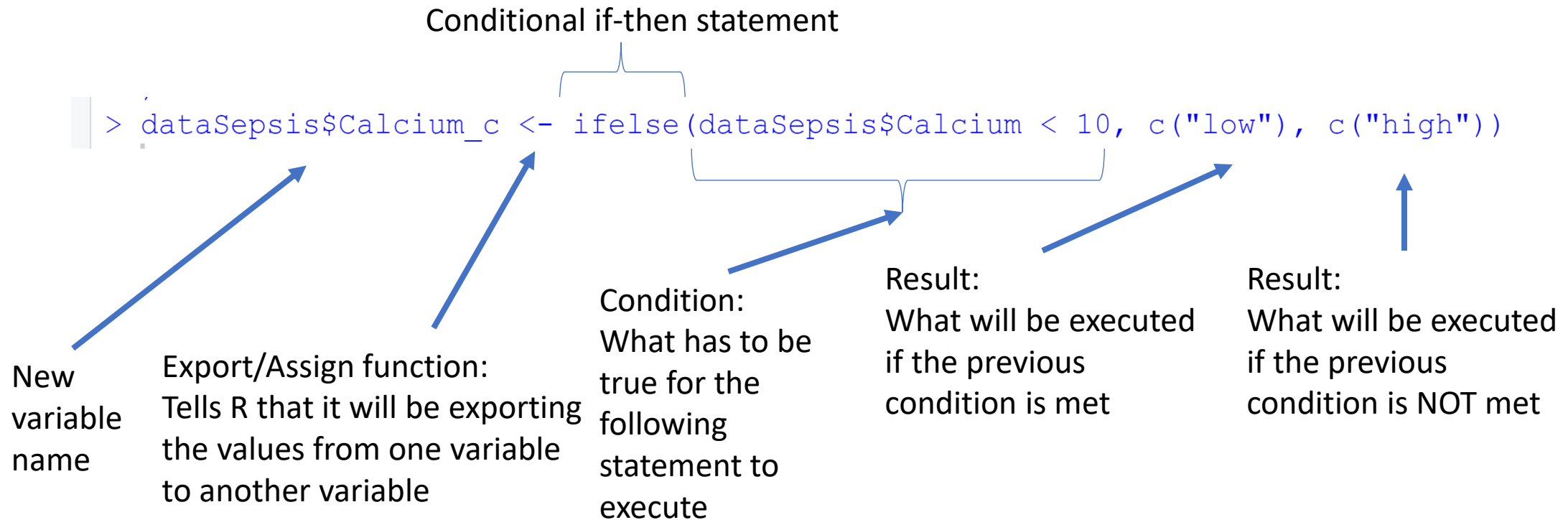
Variable 2 to be added

\*Calculations can contain multiple variables and multiple operators and functions enclosed in parentheses. See R technical documentation for additional function options that can be used in calculations.





# Populating values of a variable using an if-then statement



# Combining Multiple Character Variables

```
> dataSepsis$UnitPairID <- paste(dataSepsis$Unit1,dataSepsis$Unit2,sep=",")
```

New variable  
name

Export/Assign function:  
Tells R that it will be  
exporting the values from  
one variable to another  
variable

Concatination  
function

Variables to be  
combined

Separator (the symbol or  
character that will be used  
to separate the values  
within the new variable)



# Taking Only a Portion of a Character String

```
> dataSepsis$SubUnitPairID <- substr(dataSepsis$UnitPairID,1,2)
```

New variable  
name

Export/Assign function:  
Tells R that it will be  
exporting the values from  
one variable to another  
variable

Substring  
function

Variable we are  
taking the portion of  
the characters from

Which  
position  
character  
to start  
with

Which  
position  
character to  
stop at

